Geometria Computacional

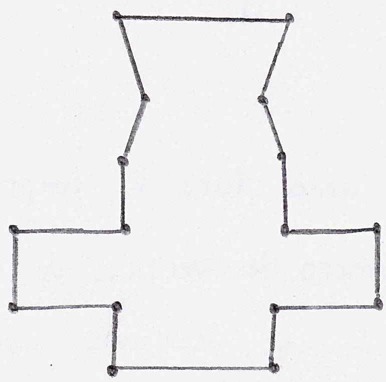
MAC 0331

Lista 4

$\boxed{8}$ Um polígono pode ser monótono em relação a precisamente uma única direção?

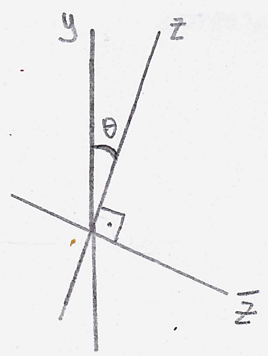
Sim.

O polígono abaixo é monótono somente na direção do eixo y:

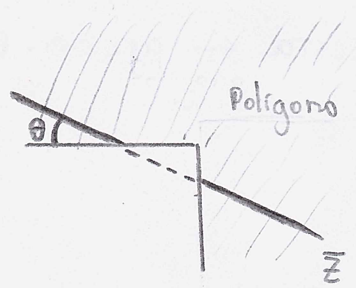


O segredo são os ângulos retos.

Suponha que o polígono acima seja monótono em alguma direção z, em que $z = y + \theta$, $\theta \in \left]-\frac{\pi}{2}, \frac{\pi}{2}\right[$.



O polígono perderá sua monotonidade na região de algum dos ângulos retos



Observe que qualquer θ produziria o mesmo efeito

(Poderia ser espelhado do outro lado)

Os ângulos retos só falham no caso da direção x, então adicionamos aqueles ângulos refletos na parte superior, tornando o polígono exclusivamente y-monótono.

10 Não foi feito ''

15 Considere um polígono particionado, representado pela ED vista em aula, que chamamos de listas de arestas duplamente ligadas (com seus registros para vértices, arestas e faces).

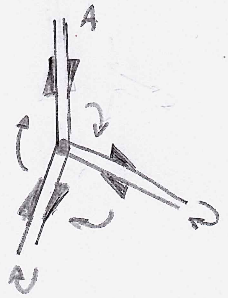
(a) Escreva um algoritmo que, dada uma face f, imprime todos os vértices desta face em tempo linear no número de vértices de f.

Como temos nossos registros de faces, devemos apenas percorrer a face contando os vértices.

```
N_VERTICES (f):

    ARESTA ← f          ▷ Considerando que a face é armazenada
                          por uma de suas arestas
    C ← 1

    enquanto aresta.v != f.u faça:

        C ← c+1

        aresta ← aresta.prox

    devolva c
```

(b) Escreva um algoritmo que, dado um vértice v, obtem todos os vértices adjacentes a v em tempo linear no número de arestas incidentes a v.

Considerando que o vértice v é armazenado com umas das arestas que tem uma ponta em v, devemos percorrer as arestas "radialmente", passandos pelas gêmeas:



Percorre A, vai pra próxima,
vai pra gêmea, vai pra próxima,
vai pra gêmea, vai pra próxima, ...

```
ADJACENTES (v):

    ARESTA ← v
    ADJ ← ∅
    se  v = ARESTA.U :
        ADJ ← ADJ ∪ {ARESTA.v}
        ARESTA ← ARESTA.TWIN
    ARESTA ← ARESTA. PROX            ▷ v é o vértice final
    enquanto  ARESTA ≠ V  e  ARESTA.TWIN ≠ V
        se  v = ARESTA.U
            ADJ ← ADJ ∪ {ARESTA.v}
            ARESTA ← ARESTA.TWIN
        ARESTA = ARESTA. PROX
    Devolve ADJ
```